PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-307840
(43)Date of publication of application : 17.11.1998

(51)Int.CI. G06F 17/30

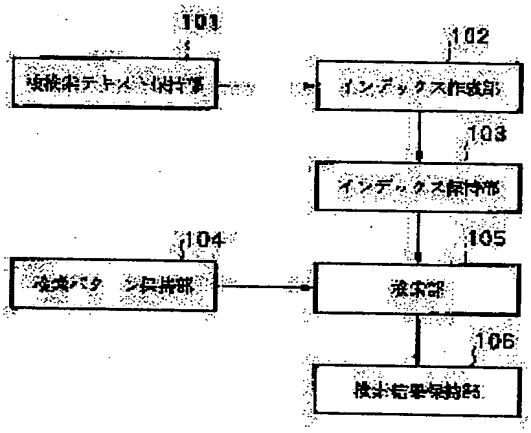
(21)Application number : 09-119869
(22)Date of filing : 09.05.1997

(71)Applicant : CANON INC
(72)Inventor : ITO SHIRO
IKEDA YUJI

BEST AVAILABLE COPY

(54) INFORMATION PROCESSOR AND ITS METHOD

(57)Abstract:
PROBLEM TO BE SOLVED: To provide an information processor and its method capable of suppressing the increment of index keys to be used for the retrieval of text data and improving retrieving speed.
SOLUTION: A retrieved text storing part 101 stores text data and an index preparing part 102 prepares indexes related to the positions of character strings constituting text data based on character strings appearing more than the prescribed number of times out of character strings constituting the stored text data. A retrieving part 105 retrieves text data having an inputted retrieving pattern by using these prepared indexes and outputs the retrieved result to a retrieved result storing part 106.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（Ａ）**　(11)特許出願公開番号

特開平10－307840

(43)公開日　平成10年(1998)11月17日

(51)Int.Cl.[6]	識別記号	ＦＩ	
Ｇ０６Ｆ　17/30		Ｇ０６Ｆ　15/413	３１０Ｂ
		15/40	３７０Ａ
		15/401	３１０Ａ

審査請求　未請求　請求項の数16　ＯＬ　（全 15 頁）

(21)出願番号　特願平9－119869

(22)出願日　平成９年(1997)５月９日

(71)出願人　000001007
キヤノン株式会社
東京都大田区下丸子３丁目30番２号

(72)発明者　伊藤　史朗
東京都大田区下丸子３丁目30番２号　キヤノン株式会社内

(72)発明者　池田　裕治
東京都大田区下丸子３丁目30番２号　キヤノン株式会社内

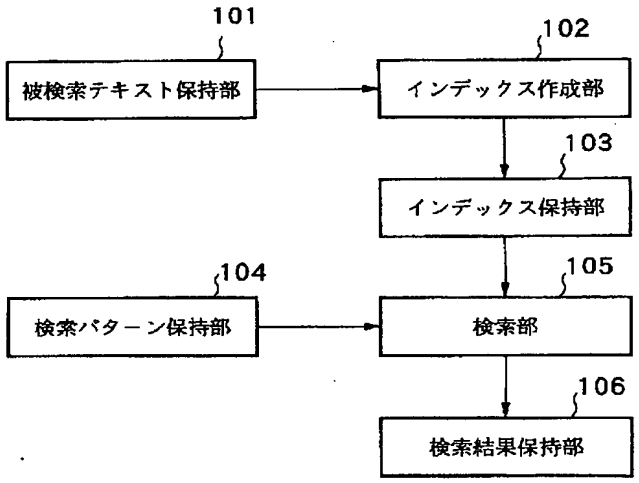
(74)代理人　弁理士　大塚　康徳　（外１名）

(54)【発明の名称】　情報処理装置及びその方法

(57)【要約】

【課題】　テキストデータの検索に用いるインデックスのキーの増大を抑えるとともに、検索速度を向上することができる情報処理装置及びその方法を提供する。

【解決手段】　テキストデータを被検索テキスト保持部１０１に保持し、保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関するインデックスをインデックス作成部１０２で作成する。作成されたインデックスを用いて、入力された検索パターンを有するテキストデータを検索部１０５で検索する。そして、検索結果を検索結果保持部１０６に出力する。



1

【特許請求の範囲】

【請求項１】　テキストデータを検索する情報処理装置であって、

テキストデータを保持する保持手段と、

前記保持手段で保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成手段と、

前記作成手段で作成された位置情報を用いて、入力された検索パターンを有するテキストデータを検索する検索手段と、

前記検索手段による検索結果を出力する出力手段とを備えることを特徴とする情報処理装置。

【請求項２】　前記作成手段は、前記所定回数以上出現する文字列と所定の位置関係を有する文字列の位置に関する位置情報を作成することを特徴とする請求項１に記載の情報処理装置。

【請求項３】　前記作成手段は、前記所定回数以上出現する文字列に前接する文字列の位置に関する位置情報を作成することを特徴とする請求項１に記載の情報処理装置。

【請求項４】　前記作成手段は、前記所定回数以上出現する文字列に後接する文字列の位置に関する位置情報を作成することを特徴とする請求項１に記載の情報処理装置。

【請求項５】　前記テキストデータを構成する文字列は、１文字を含むことを特徴とする請求項１に記載の情報処理装置。

【請求項６】　前記作成手段は、前記テキストデータ中で、前記所定回数以上出現する文字列以外の文字列の位置に関する位置情報を作成し、該位置情報を一括して管理することを特徴とする請求項１に記載の情報処理装置。

【請求項７】　テキストデータを管理する情報処理装置であって、

入力されたテキストデータを構成する文字列の内、所定回数以上出現する文字に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成手段と、

前記作成手段で作成された位置情報と前記テキストデータを対応づけて管理する管理手段とを備えることを特徴とする情報処理装置。

【請求項８】　テキストデータを検索する情報処理方法であって、

テキストデータを記憶媒体に保持する保持工程と、

前記保持工程で前記記憶媒体に保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程と、

前記作成工程で作成された位置情報を用いて、入力された検索パターンを有するテキストデータを検索する検索工程と、

前記検索工程による検索結果を出力する出力工程とを備えることを特徴とする情報処理方法。

【請求項９】　前記作成工程は、前記所定回数以上出現する文字列と所定の位置関係を有する文字列の位置に関する位置情報を作成することを特徴とする請求項８に記載の情報処理方法。

【請求項１０】　前記作成工程は、前記所定回数以上出現する文字列に前接する文字列の位置に関する位置情報を作成することを特徴とする請求項８に記載の情報処理方法。

【請求項１１】　前記作成工程は、前記所定回数以上出現する文字列に後接する文字列の位置に関する位置情報を作成することを特徴とする請求項８に記載の情報処理方法。

【請求項１２】　前記テキストデータを構成する文字列は、１文字を含むことを特徴とする請求項８に記載の情報処理方法。

【請求項１３】　前記作成工程は、前記テキストデータ中で、前記所定回数以上出現する文字列以外の文字列の位置に関する位置情報を作成し、該位置情報を一括して管理することを特徴とする請求項８に記載の情報処理方法。

【請求項１４】　テキストデータを管理する情報処理方法であって、

入力されたテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程と、

前記作成工程で作成された位置情報と前記テキストデータを対応づけて記憶媒体に管理する管理工程とを備えることを特徴とする情報処理方法。

【請求項１５】　テキストデータを検索する情報処理のプログラムコードが格納されたコンピュータ可読メモリであって、

テキストデータを記憶媒体に保持する保持工程のプログラムコードと、

前記保持工程で前記記憶媒体に保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程のプログラムコードと、

前記作成工程で作成された位置情報を用いて、入力された検索パターンを有するテキストデータを検索する検索工程のプログラムコードと、

前記検索工程による検索結果を出力する出力工程のプログラムコードとを備えることを特徴とするコンピュータ可読メモリ。

【請求項１６】　テキストデータを管理する情報処理の

プログラムコードが格納されたコンピュータ可読メモリであって、
入力されたテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程のプログラムコードと、
前記作成工程で作成された位置情報と前記テキストデータを対応づけて記憶媒体に管理する管理工程のプログラムコードとを備えることを特徴とするコンピュータ可読メモリ。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、テキストデータを検索する情報処理装置及びその方法に関するものである。

【０００２】

【従来の技術】文書中の全てのテキストデータを対象として与えられた検索パターンを含むテキストデータを検索する全文検索装置などの情報処理装置では、大量のテキストデータを高速に検索するために、被検索テキストデータのインデックスを予め作成し、作成したインデックスを用いて検索を行なうことが多い。こうしたインデックス方式の一つに、ｎ－ｇｒａｍインデックス方式と呼ばれる検索方式がある。ｎ－ｇｒａｍインデックス方式とは、テキストデータ中の連接するｎ文字をキーとして、キーとなる文字列の被検索テキストデータ中での存在位置を列挙した位置リストを保持するインデックスである。

【０００３】例えば、ｎを２に固定すると、文字セットＣに対して、文字列集合Ｓ＝｛ｃ｜ｃ∈Ｃ｝∪｛ｃ1ｃ2｜ｃ1，ｃ2∈Ｃ｝を定義する。文字セットＣから構成される検索対象文書Ｔにおいて、Ｔにおける文字列ｓ∈Ｓの出現回数をｎｓとし、Ｔにおいてｓがｘ回目に出現するときのｓを構成する先頭文字の位置をｐｏｓ（ｓ，ｘ）で表すとする。例えば、２－ｇｒａｍインデックスのキーｋは、ｎｓ≧１であるｓになる。そして、ｋに対する位置リストＬｋは、

Ｌｋ＝（ｐｏｓ（ｋ，１），ｐｏｓ（ｋ，２），…，ｐｏｓ（ｋ，ｎｋ））

となる。２－ｇｒａｍインデックスは、（ｋ，Ｌｋ）を全てのｋについて集めたものである。例えば、「日本対スイスの日本側当日券は本日売り切れ」のテキストデータに対する２－ｇｒａｍインデックスは図８のようになる。

【０００４】次に、Ｎ－ｇｒａｍインデックスを用いた検索処理について説明する。ここでも、ｎを２に固定した場合で説明するが、ｎの値が変わっても基本は同じである。被検索テキストデータＴから文字長１の検索パターンｑを検索する処理は、以下のようになる。ここで、ｓｉは、検索パターンのｉ番目から始まる２文字の部分文字列を示す。但し、ｓｌの長さだけは１とする。

【０００５】１．インデックスＩにおいて、検索パターンｑ中の部分文字列ｓｉ（ｉ＝２ｊ＋１，ｌが奇数のとき０≦ｊ≦（ｌ－１）／２，ｌが偶数のとき０≦ｊ≦ｌ／２－１）に関して、ｎｓｉの小さい順に応じてｉを並べた数列（ｔ(0)，ｔ(1)，…，ｔ(ｍ)）を作成する。ここで、ｍ＝（ｌ－１）／２（ｌが奇数の場合）、ｍ＝ｌ／２－１（ｌが偶数の場合）になる。

【０００６】２．ｎｓｔ(0)＝０ならば検索処理は終了する。検索パターンは存在しない。

３．ｋｊ＝ｓｔ(j)（０≦ｊ≦ｍ）とする。

４．Ｒ(0)＝｛（ｐ－ｔ(0)＋１）｜ｐ∈Ｌｋ0｝とする。

５．ｊ＝１，２，…，ｍまで、次の処理を繰り返す。途中でＲ(j)が空になったら検索処理は終了する。検索パターンは存在しない。

【０００７】Ｒ(j)＝｛（ｐ－ｔ(j)＋１）｜ｐ∈Ｌｋｊ，∃ｒ∈Ｒ(j-1)，ｒ＝ｐ－ｔ(j)＋１｝

６．検索処理は終了する。Ｒ(ｍ)が検索パターンの存在する位置である。例えば「当日券」の検索処理は以下のようになる。

１．ｓ1=当日、ｓ3=券であるから、ｎｓ1＝１、ｎｓ3＝１であり、ｔ(0)＝１、ｔ(1)＝３となる。

【０００８】２．ｎｓ1＝１で、０ではないので処理を続ける。

３．ｋ0＝当日、ｋ1＝券とする。

４．Ｌｋ0＝(11)であるから、Ｒ(0)＝(11)となる。

５．Ｌｋ1＝(13)であるから、１１＝（１３－３＋１）を満足し、Ｒ(1)＝(11)とする。

【０００９】６．これにより、(11)が検索結果となる。Ｎ－ｇｒａｍインデックスのｎの値を大きくすると、上記検索処理手順の第５ステップでの比較演算回数が減少するので、検索にかかる時間が短縮される。一方、ｎを大きくすると、インデックスサイズが増大したり、インデックスの作成時間が増大するという問題点がある。比較演算回数が増える要因の一つに、検索に使用するキーの出現回数（そのキーの位置リストの要素数）が大きいことがある。そこで、インデックスサイズの増大を抑えながら、検索処理時間の短縮を図るために、キーの出現回数が大きくなるようなキーについてのみｎの値を大きくするという方法がある。

【００１０】こうした方法の一つとして、菅谷他：「ｎ－ｇｒａｍ型大規模全文検索方式の開発ーインクリメンタル型ｎ－ｇｒａｍインデクス方式ー」（第５３回情報処理学会全国大会論文集，３，ｐｐ．２３５－２３６）で説明されているインクリメンタル法がある。このインクリメンタル法では、ｎ－ｇｒａｍのキーの出現回数が閾値を越えたら、そのキーの最後に１文字追加した（ｎ＋１）－ｇｒａｍを作成して、それらをキーに追加する。例えば、キーの集合をＫ、閾値をｔａとしたとき、キーｋ∈Ｋ（出現回数をｎｋとする）に対してｎｋ＞ｔａ

となったら、K’＝K∪｛s｜s＝ｃａｔ（ｋ，ｃ），ｃ∈K｝を新しいキー候補集合とする。ここで、ｃａｔ（ｓ，ｃ）は、文字列ｓの後ろに文字ｃを加えた文字列を示す。このインクリメンタル法を用いて、「日本対スイスの日本側当日券は本日売り切れ」のテキストデータに対して閾値を１としてインデックスを作成すると図９のようになる。

【００１１】別の方法として、福島他「高速全文検索のためのフレキシブル文字列インバージョン法（１）方式概要」（第５３回情報処理学会全国大会論文集，３，ｐｐ．２３９－２４０）で説明されているフレキシブル文字列インバージョン法がある。このフレキシブル文字列インバージョン法では、ｎ－ｇｒａｍのキーの出現回数が閾値を越えた場合に、（ｎ＋１）－ｇｒａｍを作成する代わりに、キーの後（前）に連接する文字にハッシングを施し、得られたハッシュ値と元のキーを組み合わせたものを新たなキーとする。例えば、キーの集合をK、閾値をｔｂとしたとき、キーｋ∈Kに対してｎｋ＞ｔｂとなったら、K’＝K∪｛ｈ｜ｈ＝ｋｅｙ（ｋ，ｈａｓｈ(c)），ｃ∈K｝を新しいキー候補集合とする。ここで、ｈａｓｈ(c)は、文字ｃにハッシュを施したハッシュ値を示す。また、ｋｅｙ（ｓ，ｉ）は、文字列ｓと整数値ｉとで構成されるキーを示す。このフレキシブル文字列インバージョン法を用いて、「日本対スイスの日本側当日券は本日売り切れ」のテキストデータに対して閾値を１としてインデックスを作成すると図１０のようになる。尚、図中＃で示している数値がハッシュ値である。

【００１２】

【発明が解決しようとする課題】しかしながら、上記従来のインクリメンタル法を用いた情報処理装置では、閾値を越えたキーに対して任意の文字を付け加えて新しいキーを作成するためキーの数が増えすぎ、ひいてはインデックスサイズを増大させるという問題があった。

【００１３】また、上記従来のフレキシブル文字列インバージョン法を用いた情報処理装置では、ハッシュ値と組み合わせたキーを用いることで検索に使用するキーの位置リスト要素数は削減されるものの、組み合わせ演算の回数を削減できず、ひいては検索時間が十分に短縮されないという問題があった。この問題は、次の例をみるとわかりやすい。

【００１４】今、アルファベットだけを対象とした場合に、キー「あ」の出現回数が閾値を越えたとする。この場合、インクリメンタル法では、新たにキー「ああ」、「あい」、「あう」、…、「あん」が作成される。このように、インクリメンタル法では、新しく作成されるキーの数が大きくなる。一方、フレキシブル文字列インバージョン法では、ハッシュ値が０と１になるようにハッシングを施したとすると、キー「あ＃０」、「あ＃１」が新しく作成されるだけで増大するキーの数を小さく抑

えられる。次に、「あいう」という検索パターンで検索することを考える。インクリメンタル法では、検索に使用するキーが「あい」と「う」となり、この二つの組み合わせ演算だけで検索が可能である。一方、フレキシブル文字列インバージョン法では、「あ＃０」、「い」、「う」の三つの組み合わせ演算が必要になる（ここでは、「い」のハッシュ値が０になると仮定している）。なぜなら、キー「あ＃０」だけでは、「あ」の次の文字が「い」である保障はないからである。

【００１５】本発明は上記の問題に鑑みてなされたものであり、テキストデータの検索に用いるインデックスのキーの増大を抑えるとともに、検索速度を向上することができる情報処理装置及びその方法を提供することを目的とする。

【００１６】

【課題を解決するための手段】上記の目的を達成するための本発明による情報処理装置は以下の構成を備える。即ち、テキストデータを検索する情報処理装置であって、テキストデータを保持する保持手段と、前記保持手段で保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成手段と、前記作成手段で作成された位置情報を用いて、入力された検索パターンを有するテキストデータを検索する検索手段と、前記検索手段による検索結果を出力する出力手段とを備える。

【００１７】また、好ましくは、前記作成手段は、前記所定回数以上出現する文字列と所定の位置関係を有する文字列の位置に関する位置情報を作成する。また、好ましくは、前記作成手段は、前記所定回数以上出現する文字列に前接する文字列の位置に関する位置情報を作成する。また、好ましくは、前記作成手段は、前記所定回数以上出現する文字列に後接する文字列の位置に関する位置情報を作成する。

【００１８】また、好ましくは、前記テキストデータを構成する文字列は、１文字を含む。また、好ましくは、前記作成手段は、前記テキストデータ中で、前記所定回数以上出現する文字列以外の文字列の位置に関する位置情報を作成し、該位置情報を一括して管理する。上記の目的を達成するための本発明による情報処理装置は以下の構成を備える。即ち、テキストデータを管理する情報処理装置であって、入力されたテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成手段と、前記作成手段で作成された位置情報と前記テキストデータを対応づけて管理する管理手段とを備える。

【００１９】上記の目的を達成するための本発明による情報処理方法は以下の構成を備える。即ち、テキストデータを検索する情報処理方法であって、テキストデータ

を記憶媒体に保持する保持工程と、前記保持工程で前記記憶媒体に保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程と、前記作成工程で作成された位置情報を用いて、入力された検索パターンを有するテキストデータを検索する検索工程と、前記検索工程による検索結果を出力する出力工程とを備える。

【００２０】上記の目的を達成するための本発明による情報処理方法は以下の構成を備える。即ち、テキストデータを管理する情報処理方法であって、入力されたテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程と、前記作成工程で作成された位置情報と前記テキストデータを対応づけて記憶媒体に管理する管理工程とを備える。

【００２１】上記の目的を達成するための本発明によるコンピュータ可読メモリは以下の構成を備える。即ち、テキストデータを検索する情報処理のプログラムコードが格納されたコンピュータ可読メモリであって、テキストデータを記憶媒体に保持する保持工程のプログラムコードと、前記保持工程で前記記憶媒体に保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程のプログラムコードと、前記作成工程で作成された位置情報を用いて、入力された検索パターンを有するテキストデータを検索する検索工程のプログラムコードと、前記検索工程による検索結果を出力する出力工程のプログラムコードとを備える。

【００２２】上記の目的を達成するための本発明によるコンピュータ可読メモリは以下の構成を備える。即ち、テキストデータを管理する情報処理のプログラムコードが格納されたコンピュータ可読メモリであって、入力されたテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する作成工程のプログラムコードと、前記作成工程で作成された位置情報と前記テキストデータを対応づけて記憶媒体に管理する管理工程のプログラムコードとを備える。

【００２３】

【発明の実施の形態】以下、図面を参照して本発明の好適な実施形態を詳細に説明する。図１は本発明の実施形態に係る情報処理装置の機能構成を示すブロック図である。図１において、１０１は被検索テキスト保持部であり、被検索テキストデータを保持する。１０２はインデックス作成部であり、被検索テキスト保持部１０１に保持されている被検索テキストデータに対して、キー（テキストデータを構成する文字あるいは文字列）集合に属するキーに対して被検索テキストデータ中での当該キー

の出現位置を列挙したインデックスを作成する。また、出現回数が基準以上のキーに１文字を付与したキー候補の中、被検索テキストデータ中での出現回数が別の基準以上のキー候補をキー集合に加えて、同様に被検索テキストデータ中での当該キーの出現位置を列挙したインデックスを作成する。１０３はインデックス保持部であり、インデックス作成部１０２で作成したインデックスを保持する。

【００２４】１０４は検索パターン保持部であり、被検索テキストデータから検索するパターンを保持する。１０５は検索部であり、インデックス保持部１０３に保持されているインデックスを用いて、検索パターン保持部１０４に保持されている検索パターンを被検索テキストデータ中から検索する。１０６は検索結果保持部であり、検索部１０５による検索結果を保持する。

【００２５】次に本発明の実施形態の情報処理装置の構成について、図２を用いて説明する。図２は本発明の実施形態の情報処理装置の構成を示すブロック図である。図２において、２０１はＣＰＵであり、後述する手順を実現するプログラムに従って動作する。２０２はＲＡＭであり、被検索テキスト保持部１０１、検索パターン保持部１０４、検索結果保持部１０６と上記プログラムの動作に必要な記憶領域とを提供する。２０３はＲＯＭであり、後述する手順を実現するプログラムを保持する。２０４はディスク装置であり、インデックス保持部１０３を実現する。２０５は情報処理装置の各種構成要素を相互に接続するバスである。２０６はキーボード及びマウスからなる入力装置であり、検索キーを入力する。２０７は例えば、ＣＲＴ、ＬＣＤ等の出力装置であり、検索結果を表示する。

【００２６】以下、説明していく本発明の実施形態で実行される処理は、インデックスを作成するインデックス作成処理と文字列を検索する検索処理の２つに大きく分かれる。まず、インデックス作成処理によって作成されるインデックスを保持するインデックス保持部１０３の詳細な構成について、図１１を用いて説明する。

【００２７】図１１は本発明の実施形態のインデックス保持部の詳細な構成を示す図である。図１１において、１１０１は拡張コード領域１１０３に示される拡張コードに１文字を後接したキーが被検索テキストデータ中に出現する回数を保持する出現回数領域である。この出現回数領域では、行が拡張コードに対応し、列が後接する文字に対応する２次元の出現回数テーブルを保持する。但し、０行目については元になるキーがなく、各列が示す文字だけのキーの出現回数を示す。

【００２８】１１０２は、拡張コードに１文字を後接したキーに対し、位置リストを保持する際のキーコードを保持するキーコード領域である。このキーコード領域では、行が拡張コードに対応し、列が後接する文字に対応する２次元のキーコードテーブルを保持する。但し、０

行目については元になるキーがなく、各列が示す文字だけのキーのキーコードを示す。また、０列目については元になるキーの位置リストから、文字を後接して作成した位置リストに含まれる要素を除いて作成する位置リストに対するキーコードを保持する。

【００２９】１１０３は、拡張コードに１文字を後接したキーをさらに1文字拡張する場合の管理を行なう拡張コードを保持する拡張コード領域である。この拡張コード領域では、行が拡張コードに対応し、列が後接する文字に対応する２次元の拡張コードテーブルを保持する。但し、０行目については元になるキーがなく、各列が示す文字だけのキーの拡張コードを示す。

【００３０】１１０４は、キーコードが示すキーに対する位置リストを保持する位置リスト領域である。キーコード値の昇順に並んでいるので、キーコード値から容易に位置リストへアクセスできる。以下の説明では、出現回数テーブルにおけるｎ行の文字ｃに対応する列の値をＣ（ｎ，ｃ）で表し、キーコードテーブルにおけるｎ行の文字ｃに対応する列の値をＫ（ｎ，ｃ）で表し、キーコードテーブルにおけるｎ行の０列目の値をＫ０（ｎ）で表し、拡張コードテーブルにおけるｎ行の文字ｃに対応する列の値をＥ（ｎ，ｃ）で表し、キーコードＫに対する位置リストをＬ（ｋ）で表す場合もある。

【００３１】次に、インデックス作成部１０２で実行されるインデックス作成処理について、図３を用いて説明する。図３は本発明の実施形態で実行されるインデックス作成処理を示すフローチャートである。まず、ステップＳ３０１では、出現回数テーブル、キーコードテーブル、拡張コードテーブルの全要素を全て０に初期化する。また、処理の対象となっている文字の位置を示すカウンタＣを１に初期化する。次に、ステップＳ３０２では、ポインタＰの初期化を行なう。ポインタＰは、処理の対象となっている文字を指し示すもので、これを被検索テキストデータの先頭文字を指すように初期化する。ステップＳ３０３では、ポインタｐが被検索テキストデータの最後に達したか否かを判定する。最後に達している場合（ステップＳ３０３でＹＥＳ）、インデックス作成処理を終了する。一方、最後に達していない場合（ステップＳ３０３でＮＯ）、ステップＳ３０４に進む。

【００３２】ステップＳ３０４では、引数（０，ｐ，１）による登録処理を行なう。尚、この登録処理の詳細については、図４を用いて後述する。また、（０，ｐ，１）は図４のフローチャートを呼び出すための引数である。ステップＳ３０５では、カウンタｃの値を１増やす。ステップＳ３０６では、ポインタｐが次の文字を指し示すようポインタｐを進め、ステップＳ３０３に戻る。

【００３３】次に、上述した図３のステップＳ３０４における引数（ｎ，ｐ，ｓ）への登録処理の詳細について、図４を用いて説明する。図４は本発明の実施形態で

実行される登録処理の詳細を示すフローチャートである。まず、ステップＳ４０１では、ポインタｐが指し示す文字をｃとする。ステップＳ４０２では、出現回数テーブルにおいて、ｎ行の文字ｃに対応する列Ｃ（ｎ，ｃ）の値を１増やす。ステップＳ４０３では、引数（ｎ，ｃ，ｓ）に対する新キー作成処理を行なう。尚、この新キー作成処理の詳細については、図５を用いて後述する。ステップＳ４０４では、キーコードテーブルにおいて、ｎ行の文字ｃに対応する列Ｋ（ｎ，ｃ）の値を参照する。列Ｋ（ｎ，ｃ）の値が０である場合は、ステップＳ４０５に進み、キーコードテーブルにおけるｎ行の０列目の値をキーコードとして、当該キーコードに対する位置リストＬ（Ｋ０（ｎ））にカウンタＣの値を追加する。そして、ステップＳ４０７に進む。一方、列Ｋ（ｎ，ｃ）の値が正の値である場合は、ステップＳ４０６に進み、キーコードテーブルにおいて、ｎ行の文字ｃに対応する列の値をキーコードとして、当該キーコードに対する位置リストＬ（Ｋ０（ｎ），ｃ）にカウンタＣの値を追加する。そして、ステップＳ４０７に進む。

【００３４】ステップＳ４０７では、拡張コードテーブルにおいて、ｎ行の文字ｃに対応する列Ｅ（ｎ，ｃ）の値を参照する。列Ｅ（ｎ，ｃ）の値が０である場合は、（ｎ，ｐ，ｓ）への登録処理を終了する。一方、列Ｅ（ｎ，ｃ）の値が正の値である場合は、ステップＳ４０８に進む。ステップＳ４０８では、ポインタｐが次の文字を指し示すようポインタｐを進める。ステップＳ４０９では、ポインタｐが検索テキストデータの最後に達したか否かを判定する。最後に達した場合（ステップＳ４０９でＹＥＳ）、（ｎ，ｐ，ｓ）への登録処理を終了する。一方、最後に達していない場合（ステップＳ４０９でＮＯ）、ステップＳ４１０に進む。

【００３５】ステップＳ４１０では、拡張コードテーブルにおいて、ｎ行の文字ｃに対応する列Ｅ（ｎ，ｃ）の値をｍとして、引数（ｍ，ｐ，（ｓ＋１））への登録処理を再帰し、引数（ｍ，ｐ，（ｓ＋１））への登録処理を終了する。次に、上述した図４のステップＳ４０３における引数（ｎ，ｃ，ｓ）に対する新キー作成処理の詳細について、図５を用いて説明する。

【００３６】図５は本発明の実施形態で実行される新キー作成処理の詳細を示すフローチャートである。まず、ステップＳ５０１では、出現回数テーブルにおいて、ｎ行の文字ｃに対応する列Ｃ（ｎ，ｃ）の値を参照し、Ｃ（ｎ，ｃ）の値が閾値Ａを越えており、かつ拡張コードテーブルにおいて、ｎ行の文字ｃに対応する列Ｅ（ｎ，ｃ）の値を参照し、列Ｅ（ｎ，ｃ）の値が０あるという条件を満たすか否かを判定する。条件を満たす場合（ステップＳ５０１でＹＥＳ）、ステップＳ５０８に進む。一方、条件を満たさない場合（ステップＳ５０１でＮＯ）、ステップＳ５０２に進む。

【００３７】ステップＳ５０２では、ｎの値を参照し、

値が０であるか否かを判定する。値が０である場合（ステップＳ５０２でＹＥＳ）、ステップＳ５０３に進む。一方、値が０でない場合（ステップＳ５０２でＮＯ）、ステップＳ５０５に進む。ステップＳ５０３では、キーコードテーブルにおいて、ｎ行の文字ｃに対応する列Ｋ（ｎ，ｃ）の値が０であるか否かを判定する。列Ｋ（ｎ，ｃ）の値が０である場合（ステップＳ５０３でＹＥＳ）、ステップＳ５０４に進み、新しいキーコードを割り当て、割り当てた値をキーコードテーブルにおけるｎ行の文字ｃに対応する列Ｋ（ｎ，ｃ）に代入する。そして、引数（ｎ，ｃ，ｓ）に対する新キー作成処理を終了する。一方、列Ｋ（ｎ，ｃ）の値が０でない場合（ステップＳ５０３でＮＯ）、（ｎ，ｃ，ｓ）に対する新キー作成処理を終了する。

【００３８】ステップＳ５０５では、出現回数テーブルにおいて、ｎ行の文字ｃに対応する列Ｃ（ｎ，ｃ）の値を参照し、列Ｃ（ｎ，ｃ）値が閾値Ｂを越えており、かつキーコードテーブルにおいて、ｎ行の文字ｃに対応する列Ｋ（ｎ，ｃ）の値を参照し、列Ｋ（ｎ，ｃ）値が０であるという条件を満たすか否かを判定する。条件を満たす場合（ステップＳ５０５でＹＥＳ）、ステップＳ５０６に進む。一方、条件を満たさない場合（ステップＳ５０５でＮＯ）、引数（ｎ，ｃ，ｓ）に対する新キー作成処理を終了する。

【００３９】ステップＳ５０６では、新しいキーコードを割り当て、割り当てた値をキーコードテーブルにおけるｎ行の文字ｃに対応する列Ｋ（ｎ，ｃ）に代入する。ステップＳ５０７では、キーコードテーブルにおけるｎ行の０列の値をキーとする位置リスト中の各値について、その値にｓを足した値の中から、キーコードテーブルにおけるｎ行の文字ｃに対応する列の値をキーとする位置リスト中の値のいずれかに一致する値を全て削除する。次に、削除した値をステップＳ５０６で新たに割り当てたキーコードに対する位置リストに加える。そして、引数（ｎ，ｃ，ｓ）に対する新キー作成処理を終了する。

【００４０】ステップＳ５０８では、新しい拡張コードを割り当て、割り当てた値ｍを拡張コードテーブルにおけるｎ行の文字ｃに対応する列Ｅ（ｎ，ｃ）に代入する。ステップＳ５０９では、新しいキーコードを割り当て、割り当てた値をキーコードテーブルにおけるｍ行の０列に代入する。そして、キーコードテーブルにおけるｎ行の文字ｃに対応する列の値をキーとする位置リストの内容を、全て割り当てたキーコードの位置リストにコピーする。

【００４１】ステップＳ５１０では、キーコードテーブルの全ての列において、以下の処理を行なう。ｍ行の０列の値をキーとする位置リスト中の各値について、その値にｓを足した値の中からキーコードテーブルにおけるｎ行の当該列の値をキーとする位置リスト中の値のいずれかに一致する値の数を調べ、それが閾値Ｂを越えていれば一致した値を全て削除する。次に、新しいキーコードを割り当て、削除した値を割り当てたキーコードの位置リストに全て加え、キーコードテーブルにおけるｍ行の当該列の値に割り当てたキーコードを代入する。そして、引数（ｎ，ｃ，ｓ）に対する新キー作成処理を終了する。

【００４２】以上のインデックス作成処理を、例えば、「日本対スイスの日本側当日券は本日売り切れ」に対して施すと、図１１に示すようなインデックスが作成される。また、図１２は、「当日券」の「日」までのインデックス作成処理が完了した時点でのインデックスを示している。次に検索部１０５で実行される検索処理について、図６を用いて説明する。

【００４３】図６は本発明の実施形態で実行される検索処理を示すフローチャートである。まず、ステップＳ６０１では、検索パターン保持部１０４に保持されている検索パターンの長さを演算用領域１に代入する。また、演算用領域ｎに１を代入する。ステップＳ６０２では、演算用領域ｎに対する位置リスト獲得処理を行ない、獲得した位置リストを演算用配列Ａ１に格納する。尚、この位置リスト獲得処理の詳細については、図７を用いて説明する。

【００４４】ステップＳ６０３では、演算用領域ｎの示す値が演算用領域１の示す値より大きいか否かを判定する。演算用領域ｎの示す値が演算用領域１の示す値より大きい場合（ステップＳ６０３でＹＥＳ）、ステップＳ６０６に進む。一方、演算用領域ｎの示す値が演算用領域１の示す値未満である場合（ステップＳ６０３でＮＯ）、ステップＳ６０４に進む。

【００４５】ステップＳ６０４では、演算用領域ｎに対する位置リスト獲得処理を行ない、獲得した位置リストを演算用配列Ａ２に格納する。ステップＳ６０５では、演算用配列Ａ１と演算用配列２の両方に存在している値を全て取り出し、これらの値だけからなる位置リストを新たに演算用配列Ａ１に格納する。そして、ステップＳ６０３に戻る。

【００４６】ステップＳ６０６では、演算用配列Ａ１が空であるか否かを調べる。空でない場合（ステップＳ６０６でＮＯ）、ステップＳ６０７に進み、被検索テキストデータから検索パターンが検索されたことを示す値として１を検索結果保持部１０６に保持する。そして検索処理を終了する。一方、空である場合（ステップＳ６０６でＹＥＳ）、ステップＳ６０８に進み、被検索テキストデータから検索パターンが検索されなかったことを示す値として０を検索結果保持部１０６に保持する。そして、検索処理を終了する。

【００４７】次に、上述した図６のステップＳ６０２における位置リスト獲得処理の詳細について、図７を用いて説明する。図７は本発明の実施形態で実行される位置

リスト獲得処理の詳細を示すフローチャートである。まず、ステップＳ７０１では、演算用領域mの値を演算用領域ｎの値に初期化する。また、演算用領域ｋの値を０に初期化する。ステップＳ７０２では、演算用領域mの値が演算用領域ｋの値以下であるか否かを判定する。演算用領域mの値が演算用領域ｋの値未満である場合（ステップＳ７０２でＹＥＳ）、ステップ７０３に進む。一方、演算用領域mの値が演算用領域ｋの値より大きい場合（ステップＳ７０２でＮＯ）、ステップＳ７０６に進む。

【００４８】ステップＳ７０３では、拡張コードテーブルのｋ行で、検索パターンのm番目の文字に対応する列の値をｋ’に代入する。ステップＳ７０４では、ｋ’の値が０であるか否かを判定する。ｋ’の値が０である場合（ステップＳ７０４でＹＥＳ）、ステップＳ７０６に進む。一方、ｋ’の値が正の値である場合（ステップＳ７０４でＮＯ）、ステップＳ７０５に進む。ステップＳ７０５では、演算用領域ｋにｋ’の値を代入し、演算用領域mの値を１増やす。そして、ステップＳ７０２に戻る。

【００４９】ステップＳ７０６では、出現回数テーブルのｋ行で、検索パターンのm番目の文字に対応する列Ｃ（ｋ，ｃ）の値が０であるか否かを判定する。列Ｃ（ｋ，ｃ）の値が０である場合（ステップＳ７０６でＹＥＳ）、ステップＳ７０７に進み、空の位置リストを獲得した位置リストとする。そして、ステップＳ７１１に進む。一方、列Ｃ（ｋ，ｃ）の値が０でない場合（ステップＳ７０６でＹＥＳ）、ステップＳ７０８に進む。

【００５０】ステップＳ７０８では、キーコードテーブルのｋ行で、検索パターンのm番目の文字に対応する列Ｋ（ｋ，ｃ）の値が０であるか否かを判定する。列Ｋ（ｋ，ｃ）の値が０である場合（ステップＳ７０８でＹＥＳ）、ステップＳ７０９に進み、キーコードテーブルのｋ行で０列の値をキーとして、そのキーコードに対応する位置リストを取り出す。そして、その全ての要素から（ｎ－１）を引いた位置リストを獲得した位置リストとする。続いて、ステップＳ７１２では、検索処理全般で使用する演算用領域ｎにmを代入する。そして、位置リスト獲得処理を終了する。

【００５１】一方、列Ｋ（ｋ，ｃ）の値が０でない場合（ステップＳ７０８でＮＯ）、ステップＳ７１０に進み、キーコードテーブルのｋ行で、検索パターンのm番目の文字に対応する列の値をキーとして、そのキーコードに対応する位置リストを取り出す。そして、その全ての要素から（ｎ－１）を引いた位置リストを獲得した位置リストとする。続いて、ステップＳ７１１では、検索処理全般で使用する演算用領域ｎに（ｍ＋１）を代入する。そして、位置リスト獲得処理を終了する。

【００５２】次に、検索パターン「当日券」で検索する場合の検索処理の具体例について説明していく。

１．ｌ＝３、ｎ＝１を代入する。（ステップＳ６０１）

２．ｎ＝１で位置リスト獲得処理を行なう。（ステップＳ６０２）

（ａ）ｍ＝ｎ＝１、ｋ＝０を代入する。（ステップＳ７０１）

（ｂ）ｍ＜ｌなので（ステップＳ７０２でＹＥＳ）、ｋ’＝Ｅ（０，当）＝０とする。（ステップＳ７０３）

（ｃ）ｋ’＝０（ステップＳ７０４でＹＥＳ）、Ｃ（０，当）＝１（ステップＳ７０６でＮＯ）、Ｋ（０，当）＝８（ステップＳ７０８でＮＯ）なので、Ｌ(8)＝(11)から０を引いた(11)を獲得する位置リストとする。（ステップＳ７１０）

（ｄ）ｎ＝ｍ＋１＝２とする。（ステップＳ７１１）

３．Ａ1＝(11)とする。（ステップＳ６０２）

４．ｎ≦ｌなので（ステップＳ６０３でＮＯ）、位置リスト獲得処理を行なう。（ステップＳ６０４）

（ａ）ｍ＝ｎ＝２、ｋ＝０。（ステップＳ７０１）

（ｂ）ｍ＜ｌなので（ステップＳ７０２でＹＥＳ）、ｋ’＝Ｅ（０，日）＝１とする。（ステップＳ７０３）

（ｃ）ｋ’＞０（ステップＳ７０４でＮＯ）なので、ｋ＝１、ｍ＝ｍ＋１＝３とする。（ステップＳ７０５）

（ｄ）ｍ＝１（ステップＳ７０２でＮＯ）、Ｃ（１，券）＝１（ステップＳ７０６でＮＯ）、Ｋ（１，券）＝０（ステップＳ７０８でＹＥＳ）なので、Ｌ（Ｋ0(1)）＝Ｌ(9)＝(12,16)から１を引いた(11,15)を獲得する位置リストとする。（ステップＳ７０９）

（ｅ）ｎ＝ｍ＝３とする。（ステップＳ７１２）

５．Ａ2＝(11,15)とする。（ステップＳ６０４）

６．Ａ1とＡ2の両方に存在する要素をとり、Ａ1＝(11)とする。（ステップＳ６０５）

７．ｎ≦ｌなので（ステップＳ６０３でＮＯ）、位置リスト獲得処理を行なう。（ステップＳ６０４）

（ａ）ｍ＝ｎ＝３、ｋ＝０。（ステップＳ７０１）

（ｂ）ｍ≧ｌなので（ステップＳ７０２でＮＯ）、Ｃ（０，券）＝１（ステップＳ７０６でＮＯ）、Ｋ（０，券）＝１１（ステップＳ７０８でＮＯ）なので、Ｌ(11)＝(13)から２を引いた(11)を獲得する位置リストとする。（ステップＳ７１０）

（ｃ）ｎ＝ｍ＋１＝４とする。（ステップＳ７１１）

８．Ａ2＝(11)とする。（ステップＳ６０４）

９．Ａ1とＡ2の両方に存在する要素をとり、Ａ1＝(11)とする。（ステップＳ６０５）

１０．ｎ＞ｌなので（ステップＳ６０３でＹＥＳ）、Ａ1は空でないので（ステップＳ６０６でＮＯ）、検索パターンが検索される。（ステップＳ６０７）

次に別の検索パターン「日本人」で検索する場合の検索処理の具体例について説明していく。

【００５３】

１．ｌ＝３、ｎ＝１を代入する。（ステップＳ６０１）

２．ｎ＝１で位置リスト獲得処理を行なう。（ステップ

Ｓ６０２）
（ａ）ｍ＝ｎ＝１、ｋ＝０を代入する。（ステップＳ７０１）
（ｂ）ｍ＜１なので（ステップＳ７０２でＹＥＳ）、ｋ’＝Ｅ（０，日）＝１とする。（ステップＳ７０３）
（ｃ）ｋ’＞０（ステップＳ７０４でＮＯ）、ｋ＝１、ｍ＝ｍ＋１とする。
【００５４】（ステップＳ７０５）
（ｄ）ｍ＜１なので（ステップＳ７０２でＹＥＳ）、ｋ’＝Ｅ（１，本）＝０とする。（ステップＳ７０３）
（ｅ）ｋ’＝０（ステップＳ７０４でＹＥＳ）、Ｃ（１，本）＝２（ステップＳ７０６でＮＯ）、Ｋ（１，本）＝１０（ステップＳ７０８でＮＯ）なので、Ｌ(10)＝(1,8)から０を引いた(1,8)を獲得する位置リストとする。（ステップＳ７１０）
（ｆ）ｎ＝ｍ＋１＝３とする。（ステップＳ７１１）
３．Ａ1＝(1,8)とする。（ステップＳ６０２）
４．ｎ≦１なので（ステップＳ６０３でＮＯ）、位置リスト獲得処理を行なう。（ステップＳ６０４）
（ａ）ｍ＝ｎ＝３、ｋ＝０。（ステップＳ７０１）
（ｂ）ｍ≧１なので（ステップＳ７０２でＮＯ）、Ｃ（０，人）＝０（ステップＳ７０６でＹＥＳ）、空リスト()を獲得する位置リストとする。（ステップＳ７０７）
（ｃ）ｎ＝ｍ＋１＝４とする。（ステップＳ７１１）
５．Ａ2＝()とする。（ステップＳ６０４）
６．Ａ1とＡ2の両方に存在する要素をとり、Ａ1＝()とする。（ステップＳ６０５）
７．ｎ＞１なので（ステップＳ６０３でＹＥＳ）、Ａ1は空であるので（ステップＳ６０６でＹＥＳ）、検索パターンは検索されない。（ステップＳ６０８）
以上のように、本実施形態の情報処理装置の検索によれば、インクリメンタル法などを利用した従来の情報処理装置の検索に比べてキーの数を削減することができる。上述の具体例において、従来のインクリメンタル法では「日本」、「日券」、「日売」の３つのキーが作成されるのに対し、本実施形態では「日本」、「日(o)」の２つとなり、キーの数が削減される。
【００５５】また、本実施形態では、作成するキーの数が削減されても、検索処理における組み合わせ演算数が大きく増えるわけではない。例えば、上述の「日本人」の検索例では、１つの閾値を用いた１回の組み合わせ演算で検索が完了している。これに対し、同様の閾値を使って、従来のフレキシブル文字列インバージョン法では、組み合わせ演算が２回必要になる。つまり、本実施形態では、従来に比べて、より高速に検索処理を実行することができる。
【００５６】以上説明したように、本実施形態によれば、文字列としての一致だけではなく、インデックスのキー数の増大を抑えるとともに組み合わせ演算数も削減

することで、インデックスサイズを抑えながら、検索時間の短縮が可能になるという効果が得られる。尚、上記実施形態においては、キーを拡張する際に全ての文字を後接した文字列で候補を作成する場合について説明したが、これに限定されるものではない。例えば、文字単独での出現回数が閾値Ｃを越える文字だけを後接する候補としてもよい。これにより、インデックス作成処理の計算量が削減される。つまり、上述した実施形態の新キー作成処理において、出現回数テーブルの０行目の値が閾値Ｃ以下の列については、各ステップでの処理を行なわなければよい。あるいは、各テーブルの列には、出現回数が閾値Ｃを越えた文字しか入れない方法でもよい。この場合、各テーブルの０行目は分離したデータ構造とする。尚、どちらの方法にしても、本実施形態では、文字の出現回数が閾値Ｃを越えた時点で、拡張するキーの再計算処理を更に加える必要がある。
【００５７】また、上記実施形態においては、キーを拡張するにあたり新しいキーの出現回数が閾値Ａを越えない場合は、残りをまとめたキーに一括して保持する場合について説明したが、これに限定されるものではない。新しいキーの出現回数が閾値Ａを越えない場合は、保持しなくてもよい。すなわち、キーコードテーブルの０列目を作成しないことになる。また、拡張する前のキーを使用すれば検索は可能である。但し、組み合わせ演算を行なう際のリストの要素数が増えるので、検索時間が増大する。また、新キー作成処理では、ステップＳ５０７とステップＳ５１０で、新しく位置リストを作成する場合には、拡張する前の元のキーの位置リストから求めればよい。
【００５８】また、上記実施形態においては、全ての文字について一括のインデックスを作成する場合について説明したが、これに限定されるものではない。ひらがな、カタカナ、漢字などの字種ごとにインデックスを分けてもよい。この場合、拡張する場合には同一字種の文字でのみ拡張してもよいし、他の字種の文字も含めて拡張してもよい。
【００５９】また、上記実施形態においては、インデックスを作成しながらキーの拡張を行なう場合について説明したが、これに限定されるものではない。一度被検索テキストデータ中でのキーの出現回数を調べて、出現回数テーブルを作成する。そして、その作成された出現回数テーブルに基づいて、キーコードテーブル、拡張コードテーブルを作成してから、位置リストの作成を進めてもよい。
【００６０】また、上記実施形態においては、拡張する前のキーのキー長を１とする場合について説明したが、これに限定されるものではない。全て２以上の一定値であってもよいし、キーの字種などによりキー長を変えてもよい。また、上記実施形態においては、拡張するキーのキー長の上限がない場合について説明したが、これに

限定されるものではない。拡張するキーのキー長に上限を設けてもよい。

【００６１】また、上記実施形態においては、キーを拡張する際に、文字を後接する場合について説明したが、これに限定されるものではない。前接したり間隔をあけた位置の文字を定めるなど任意に拡張してよい。また、上記実施形態においては、２次元のテーブルを利用してインデックスを管理する場合について説明したが、これに限定されるものではない。例えば、図１５に示すようなトライを用いても実現できる。

【００６２】図１５は本発明の他の実施形態のインデックスのデータ構造と作成例を示す図である。図１５において、１５０１はルートノードである。ルートノード１５０１は2つのデータをノード１５０２、ノード１５０３にそれぞれ保持する。ノード１５０２は、そのノード１５０２をキーとした場合のキーの出現回数を示すデータである。ノード１５０３は、そのノード１５０３をキーとした場合のキーコードを示すデータである。トライでは、上位のノードから自ノードまで辿る間の文字を付加したものが自ノードのキーとなる。従って、例えば、ノード１５０４は、キー「日本」を表す。また、ノード１５０５は、「日」を拡張したキーで、実際に作成されたキーに含まれない位置リストを保持するキーを示す。尚、位置リストは、図１１の位置リスト領域１１０４と同じになる。

【００６３】また、上記実施形態においては、図１１に示す各領域を予め十分な大きさで確保しておく場合について説明したが、これに限定されるものではない。必要に応じて領域を増大させてもよい。また、上記実施形態においては、閾値を固定した場合について説明したが、これに限定されるものではない。字種などにより異なる閾値を用いてもよい。また、被検索テキストデータの大きさなどに応じて閾値を変化させてもよい。

【００６４】また、上記実施形態においては、出現回数テーブル、キーコードテーブル、拡張コードテーブルの各テーブルの列を求める際に文字コードを使用する場合について説明したが、これに限定されるものではない。文字に対して内部的なコードを割り当て、そのコードで列を参照してもよい。また、上記実施形態においては、出現回数テーブル、キーコードテーブル、拡張コードテーブルの各テーブルや位置リストを参照する際にコードを使用する場合について説明したが、これに限定されるものではない。各テーブルのアドレスを指し示すポインタなどを使用してもよい。

【００６５】また、上記実施形態においては、被検索テキストデータ中に検索パターンが存在するか否かを検索結果として保持する場合について説明したが、これに限定されるものではない。被検索テキスト中の検索パターンの存在位置を検索結果として保持してもよい。この場合、図６で説明した検索処理のステップＳ６０６の配列Ａ1の値が検索パターンの存在位置を示すので、これを用いれば、検索パターンの存在位置を検索結果として保持することができる。

【００６６】また、上記実施形態においては、単一の被検索テキストデータに対して検索を行なう場合について説明したが、これに限定されるものではない。複数のテキストデータに対し、検索パターンが含まれるテキストデータを検索するために用いてもよいことは言うまでもない。また、上記実施形態においては、インデックス作成処理と検索処理を同一の情報処理装置で行なう場合について説明したが、これに限定されるものではない。インデックス作成処理と検索処理を異なる情報処理装置で行ってもよい。この場合の各情報処理装置の機能構成について、図１３と図１４を用いて説明する。尚、図１３に示す情報処理装置と、図１４に示す情報処理装置は、例えば、ネットワーク回線で接続され互いにデータの授受を可能とする構成になっている。また、あるいはインデックス作成処理をＣＤ－ＲＯＭ等の記憶媒体に記憶しておき、その記憶媒体を図１４に示す情報処理装置に搭載して互いにデータの授受を可能とする構成になっている。また、あるいは、検索処理をＣＤ－ＲＯＭ等の記憶媒体に記憶しておき、その記憶媒体を図１３に示す情報処理装置に搭載して互いにデータの授受を可能とする構成になっている

【００６７】図１３は本発明の他の実施形態に係る情報処理装置の機能構成を示すブロック図である。図１３において、１３０１は被検索テキスト保持部であり、被検索テキストデータを保持する。１３０２はインデックス保持部であり、被検索テキスト保持部１３０１に保持されている被検索テキストデータに対して、キー集合に属するキーに対して被検索テキストデータ中での当該キーの出現位置を列挙すると共に、キーの出現回数が基準以上の場合に、当該キーに１文字を付与したキー候補の中から、被検索テキストデータ中での出現回数が別の基準以上の場合に、当該キーをキー集合に加えて、同様に被検索テキストデータ中での当該キーの出現位置を列挙したインデックスを作成する。１３０３はインデックス保持部であり、インデックス作成部１３０２で作成したインデックスを保持する。

【００６８】図１４は本発明の他の実施形態に係る情報処理装置の機能構成を示すブロック図である。図１４において、１４０１はインデックス保持部であり、図１３に示す情報処理装置で作成されたインデックスを保持する。１４０２は検索パターン保持部であり、被検索テキストデータから検索するパターンを保持する。１４０３は検索部であり、インデックス保持部１４０１に保持されているインデックスを用いて、検索パターン保持部１４０２に保持されている検索パターンを被検索テキストデータ中から検索する。１４０４は検索結果保持部であり、検索部１４０３による検索結果を保持する。

【００６９】また、上記実施形態においては、被検索テキスト保持部１０１、検索パターン保持部１０４、検索結果保持部１０６をＲＡＭ２０２で、インデックス保持部１０２をディスク装置２０４で実現する場合について説明したが、これに限定されるものではなく、任意の記憶媒体を用いて実現してもよい。また、上記実施形態においては、各構成要素を同一の情報処理装置上で構成する場合について説明したが、これに限定されるものではなく、ネットワーク上に分散した情報処理装置などに分かれて各構成要素を構成してもよい。

【００７０】また、上記実施形態においては、プログラムをＲＯＭ２０３に保持する場合について説明したが、これに限定されるものではなく、任意の記憶媒体を用いて実現してもよい。また、同様の動作をする回路で実現してもよい。尚、本発明は、複数の機器（例えば、ホストコンピュータ、インタフェース機器、リーダ、プリンタ等）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機、ファクシミリ装置等）に適用してもよい。

【００７１】また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはＣＰＵやＭＰＵ）が記憶媒体に格納されたプログラムコードを読出し実行することによっても、達成されることは言うまでもない。

【００７２】この場合、記憶媒体から読出されたプログラムコード自体が上述した実施の形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。プログラムコードを供給するための記憶媒体としては、例えば、フロッピディスク、ハードディスク、光ディスク、光磁気ディスク、ＣＤ－ＲＯＭ、ＣＤ－Ｒ、磁気テープ、不揮発性のメモリカード、ＲＯＭなどを用いることができる。

【００７３】また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているＯＳ（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施の形態の機能が実現される場合も含まれることは言うまでもない。

【００７４】更に、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるＣＰＵなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【００７５】本発明を上記記憶媒体に適用する場合、そ

の記憶媒体には、先に説明したフローチャートに対応するプログラムコードを格納することになるが、簡単に説明すると、図１６、図１７のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。すなわち、図１６に示す、少なくとも「保持モジュール」、「作成モジュール」、「検索モジュール」および「出力モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。

【００７６】尚、「保持モジュール」は、テキストデータを記憶媒体に保持する。「作成モジュール」は、記憶媒体に保持されているテキストデータを構成する文字列の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する。「検索モジュール」は、作成された位置情報を用いて、入力された検索パターンを有するテキストデータを検索する。「出力モジュール」は、検索結果を出力する。

【００７７】また、図１７に示す、少なくとも「作成モジュール」および「管理モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。尚、「作成モジュール」は、入力されたテキストデータを構成する文字列の各文字の内、所定回数以上出現する文字列に基づいて、該テキストデータを構成する文字列の位置に関する位置情報を作成する。「管理モジュール」は、作成された位置情報とテキストデータを対応づけて記憶媒体に管理する。

【００７８】

【発明の効果】以上説明したように、本発明によれば、テキストデータの検索に用いるインデックスのキーの増大を抑えるとともに、検索速度を向上することができる情報処理装置及びその方法を提供できる。

【図面の簡単な説明】

【図１】本発明の実施形態に係る情報処理装置の機能構成を示すブロック図である。

【図２】本発明の実施形態の情報処理装置の構成を示すブロック図である。

【図３】本発明の実施形態で実行されるインデックス作成処理を示すフローチャートである。

【図４】本発明の実施形態で実行される登録処理の詳細を示すフローチャートである。

【図５】本発明の実施形態で実行される新キー作成処理の詳細を示すフローチャートである。

【図６】本発明の実施形態で実行される検索処理を示すフローチャートである。

【図７】本発明の実施形態で実行される位置リスト獲得処理の詳細を示すフローチャートである。

【図８】従来の情報処理装置におけるインデックスの概念を示す図である。

【図９】従来の情報処理装置におけるインデックスの概念を示す図である。

21

【図１０】従来の情報処理装置におけるインデックスの概念を示す図である。

【図１１】本発明の実施形態のインデックス保持部の詳細な構成を示す図である。

【図１２】本発明の実施形態のインデックス保持部の詳細な構成を示す図である。

【図１３】本発明の他の実施形態に係る情報処理装置の機能構成を示すブロック図である。

【図１４】本発明の他の実施形態に係る情報処理装置の機能構成を示すブロック図である。

【図１５】本発明の他の実施形態のインデックスのデータ構造と作成例を示す図である。

【図１６】本発明の実施形態を実現するプログラムコードを格納した記憶媒体のメモリマップの構造を示す図である。

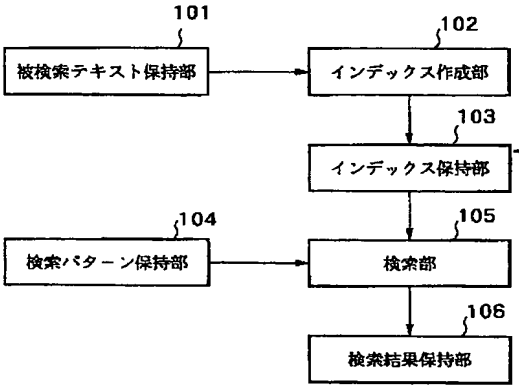
22

【図１７】本発明の実施形態を実現するプログラムコードを格納した記憶媒体のメモリマップの構造を示す図である。

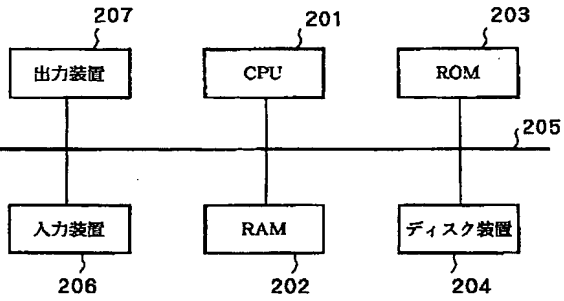
【符号の説明】

１０１　被検索テキスト保持部
１０２　インデックス作成部
１０３　インデックス保持部
１０４　検索パターン保持部
１０５　検索部
１０６　検索結果保持部

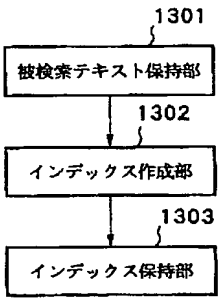
【図１】



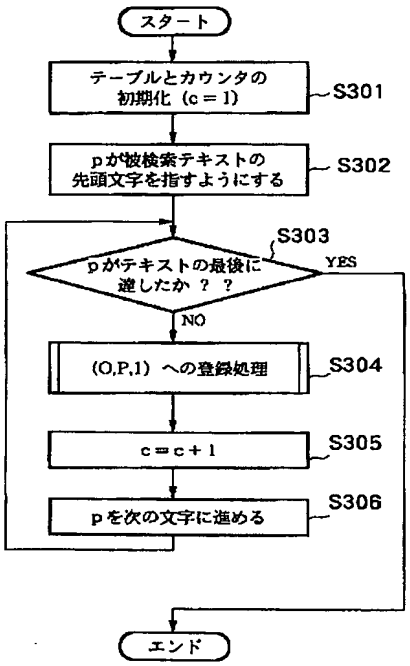
【図２】



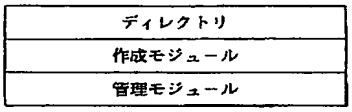
【図１３】



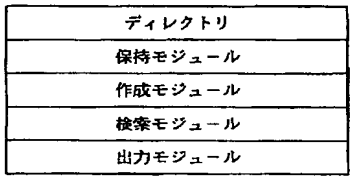
【図３】



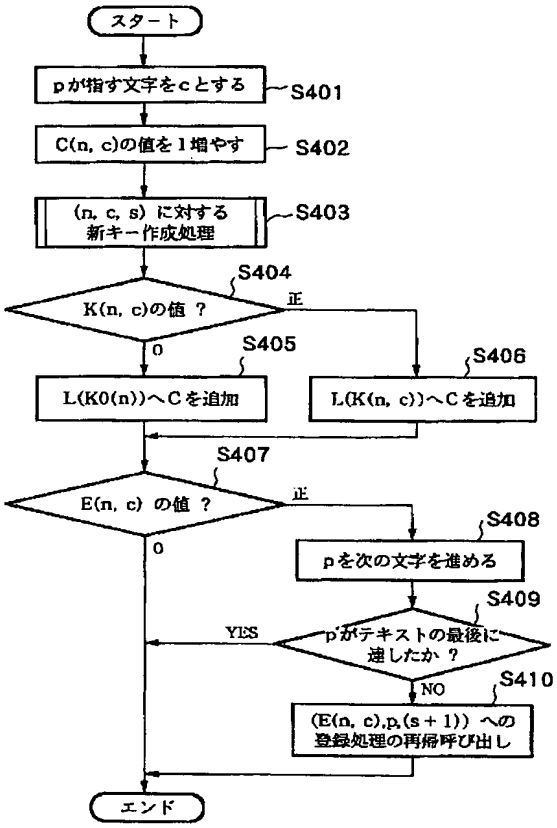
【図１７】



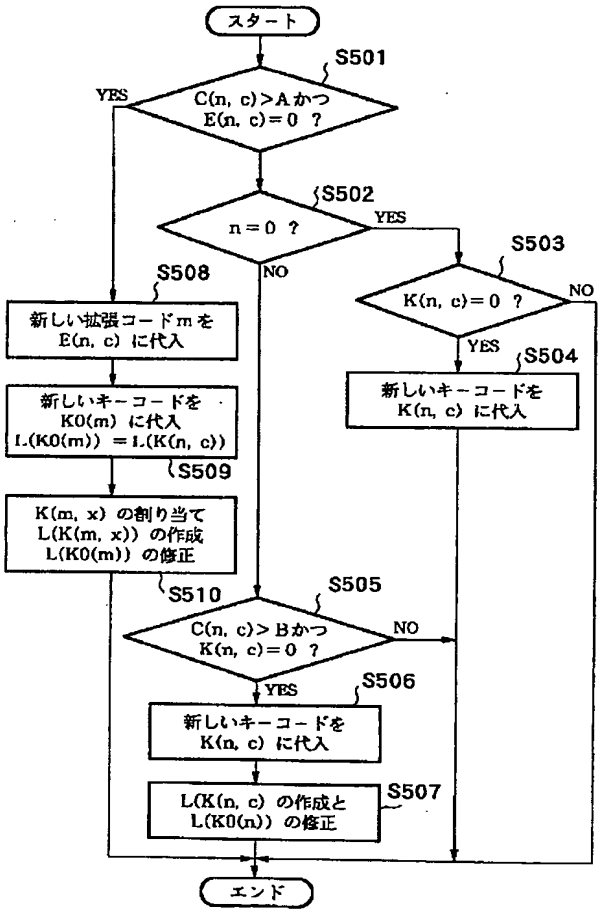
【図１６】



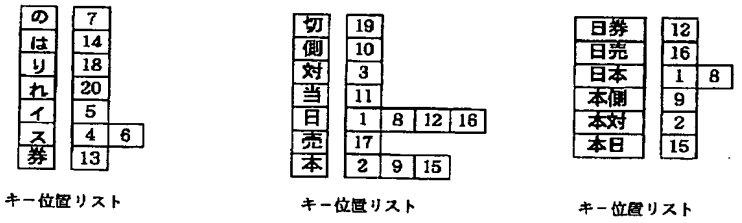
【図４】



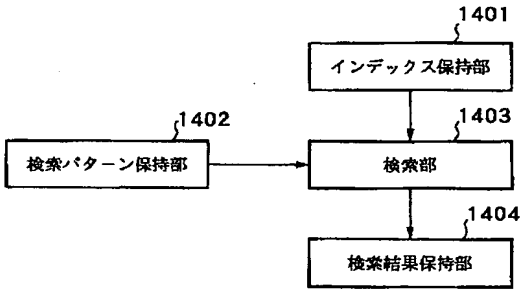
【図５】



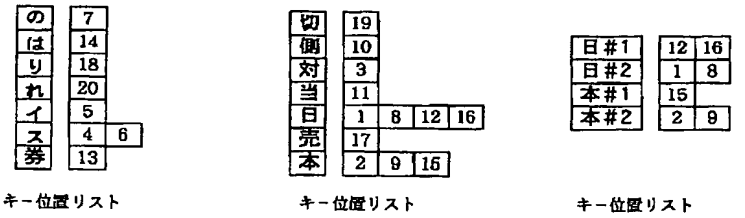
【図８】



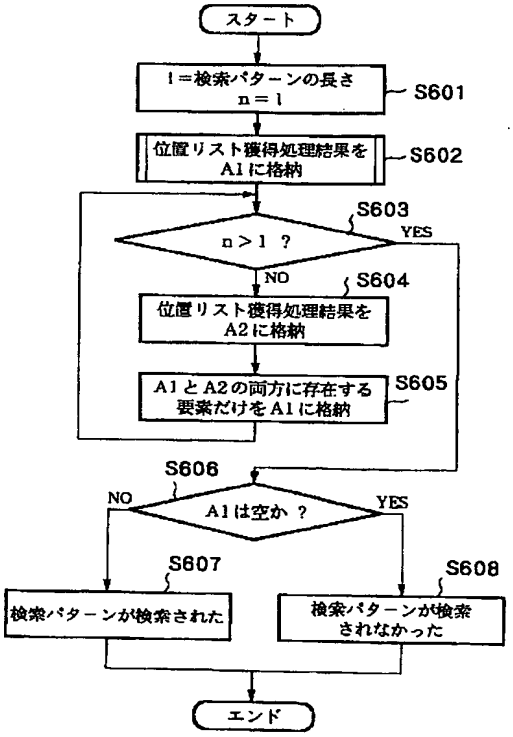
【図１４】



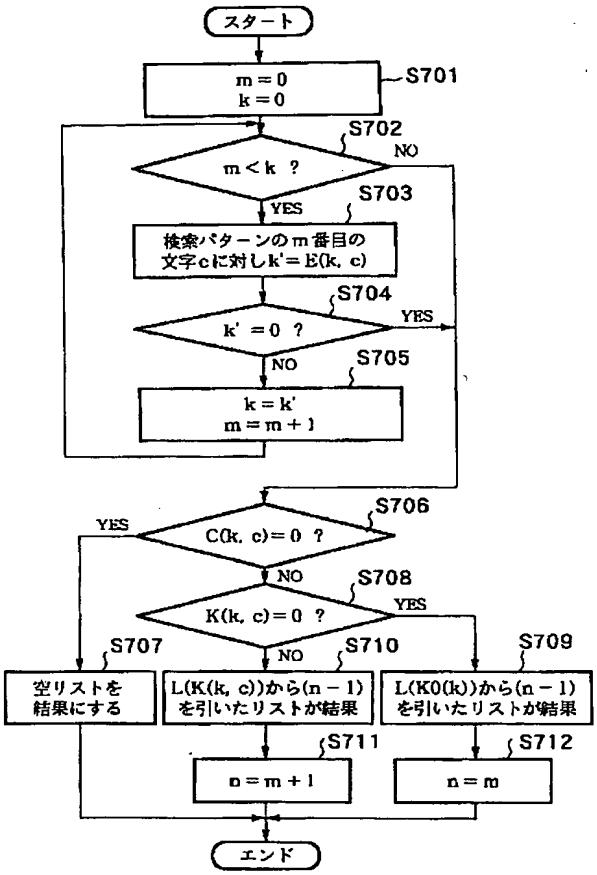
【図９】



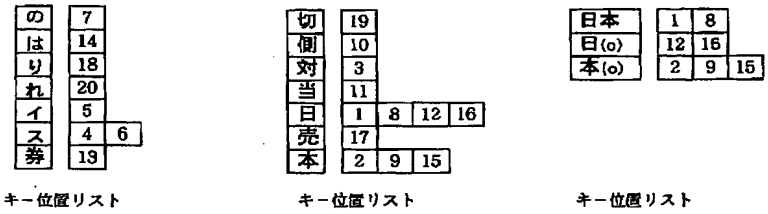
【図６】


スタート
i＝検索パターンの長さ
n＝i
S601
位置リスト獲得処理結果を
A1に格納
S602
S603
n＞1 ？
YES
NO
S604
位置リスト獲得処理結果を
A2に格納
A1とA2の両方に存在する
要素だけをA1に格納
S605
S606
NO
A1は空か ？
YES
S607
検索パターンが検索された
S608
検索パターンが検索
されなかった
エンド

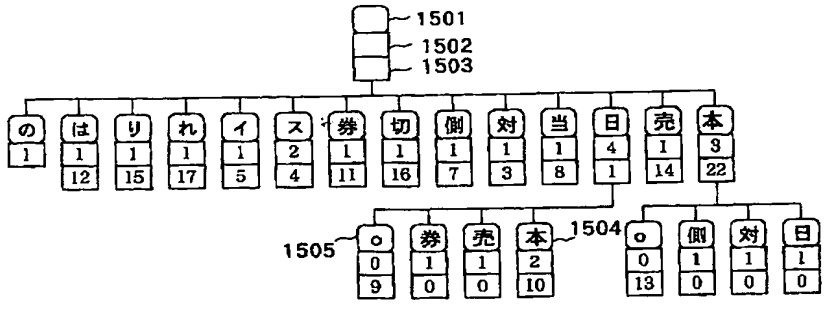
【図７】


スタート
m＝0
k＝0
S701
S702
NO
m＜k ？
YES
S703
検索パターンのm番目の
文字cに対しk'＝E(k, c)
S704
k' ＝0 ？
YES
NO
S705
k＝k'
m＝m＋1
S706
YES
C(k, c)＝0 ？
NO
S708
K(k, c)＝0 ？
YES
NO
S707
空リストを
結果にする
S710
L(K(k, c))から(n－1)
を引いたリストが結果
S709
L(K0(k))から(n－1)
を引いたリストが結果
S711
n＝m＋1
S712
n＝m
エンド

【図１０】


の 7
は 14
り 18
れ 20
イ 5
ス 4 6
券 19
キー位置リスト
切 19
側 10
対 3
当 11
日 1 8 12 16
売 17
本 2 9 15
キー位置リスト
日本 1 8
日(o) 12 16
本(o) 2 9 15
キー位置リスト

【図１５】


1501
1502
1503
の 1
は 1 12
り 1 15
れ 1 17
イ 1 5
ス 2 4
券 1 11
切 1 16
側 1 7
対 1 3
当 1 8
日 4 1
売 1 14
本 3 22
1505
o 0 9
券 1 0
売 1 0
本 2 10
1504
o 0 13
側 1 0
対 1 0
日 1 0

【図１１】

出現回数領域　1101

	あ	…	の	は	り	れ	イ	ス	券	切	側	対	当	日	売	本
0	0	…	1	1	1	1	1	2	1	1	1	1	1	4	1	3
1	0	…	0	0	0	0	0	0	1	0	0	0	0	0	1	2
2	0	…	0	0	0	0	0	0	0	0	1	1	0	1	0	0

キーコード領域　1102

	あ	…	の	は	り	れ	イ	ス	券	切	側	対	当	日	売	本
0	0	0	…	6	12	15	17	5	4	11	16	7	3	8	1	14	2
1	9	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	10
2	13	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	0

拡張コード領域　1103

	あ	…	の	は	り	れ	イ	ス	券	切	側	対	当	日	売	本
0	0	…	0	0	0	0	0	0	0	0	0	0	0	1	0	2
1	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位置リスト領域　1104

1	1	8	12	16
2	2	9	15	
3	9			
4	4	6		
5	5			
6	7			
7	10			
8	11			
9	12	16		
10	1	8		
11	13			
12	14			
13	2	9	15	
14	17			
15	18			
16	19			
17	20			

【図１２】

出現回数領域　1101

	あ	…	の	は	り	れ	イ	ス	券	切	側	対	当	日	売	本
0	0	…	1	0	0	0	1	2	0	0	1	1	1	3	0	2
1	0	…	0	0	0	0	0	0	1	0	0	0	0	0	0	2
2	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	0

キーコード領域　1102

	あ	…	の	は	り	れ	イ	ス	券	切	側	対	当	日	売	本
0	0	0	…	6	0	0	0	5	4	0	0	7	3	8	1	0	2
1	9	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	10
2	0	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	0

拡張コード領域　1103

	あ	…	の	は	り	れ	イ	ス	券	切	側	対	当	日	売	本
0	0	…	0	0	0	0	0	0	0	0	0	0	0	1	0	0
1	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	…	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位置リスト領域　1104

1	1	8	12
2	2	9	
3	3		
4	4	6	
5	5		
6	7		
7	10		
8	11		
9	12		
10	1	8	

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☑ FADED TEXT OR DRAWING
- ☑ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☑ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.